Machine Learning Engineer Nanodegree

Capstone Proposal

Christopher A. Martinez

November 12, 2018

Proposal

Domain Background

This project is based on the Google Analytics Customer Revenue Prediction competition detailed here: https://www.kaggle.com/c/ga-customer-revenue-prediction

It is worth mentioning that it is based on the first version of the data (v1) and not the current (v2).

The 80/20 rule has proven true for many businesses–only a small percentage of customers produce most of the revenue. As such, marketing teams are challenged to make appropriate investments in promotional strategies. [1]

The goal of the competition is to demonstrate how important is machine learning and data analysis for companies, by predicting the total amount spent on the Google Merchandise Store for each customer. It will be very helpful for every company to better identify and target the type of customer that is expending more money on their business given that most of the people visiting their business are not going to purchase anything, so it will be wasteful to spend resource targeting them.

Problem Statement

A very few percentage of all the people who visit a website do not buy anything, or goal here is to predict how much a visitor will spend on the google store based on some data set. By doing so we will improve our target marketing and focus it on those customers who are likely to spend more money on the site.

We need to predict the the natural log of the sum of all transactions per user using a pre given training set of ninety thousand rows by fourteen columns.

$$y_{user} = \sum_{i=1}^{n} transaction_{user_i}$$

$$target_{user} = \ln(y_{user} + 1)$$

We need to predict the the natural log of the sum of all transactions per user using a pre given training set of ninety thousand rows by fourteen columns.

We will try different machine learning algorithms on the dataset measuring their efectiveneel on the mean squared error. Once we find the best performer we will tune it using the untuned model a benchmark model. Finally we will make some predictions on the testing set.

Datasets and Inputs

We have a dataset of approximately 90 thousand rows , each corresponding to a single visit to the store[2]. Each row has the next columns:

1. fullVisitorId- A unique identifier for each user of the Google Merchandise Store.

2. channelGrouping - The channel via which the user came to the Store.

3. date - The date on which the user visited the Store.

4. device - The specifications for the device used to access the Store.

5. geoNetwork - This section contains information about the geography of the user.

6. socialEngagementType - Engagement type, either "Socially Engaged" or "Not Socially Engaged".

7. totals - This section contains aggregate values across the session.

8. trafficSource - This section contains information about the Traffic Source from which the session originated.

9. visitId - An identifier for this session. This is part of the value usually stored as the _utmb cookie. This is only unique to the user. For a completely unique ID, you should use a combination of fullVisitorId and visitId.

10. visitNumber - The session number for this user. If this is the first session, then this is set to 1.

11. visitStartTime - The timestamp (expressed as POSIX time).

12. hits - This row and nested fields are populated for any and all types of hits. Provides a record of all page visits.

13. customDimensions - This section contains any user-level or session-level custom dimensions that are set for a session. This is a repeated field and has an entry for each dimension that is set.

14. totals - This set of columns mostly includes high-level aggregate data.

There are multiple columns which contain JSON blobs of varying depth. In one of those JSON columns, totals, the sub-column ‘transactionRevenue’ contains the revenue information we are trying to predict.

Solution Statement

We will use various pre-processing techniques to generate new features and otherwise prepare the data. Following this we will split the data and test the performance of three regressors based on the RMSE :

- ADAboostregressor
- XGBregressor
- GradientBoostingRegressor

We will pick the best performer and use it as our benchmark model. We will try to beat this benchmark using hyperparameter tuning with the help of on Bayesian Optimization.

Finally we will submit our predictions to kaggle.

Benchmark Model

The best ‘ untuned’ performer of the above mentioned algorithms.

Evaluation Metrics

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2},$$

We will use the Root Mean Squared Error as our evaluation metrics, more specifically we will use scikit learn metrics module to measure the RMSE.

Project Design

We will follow the next steps towards predicting the customer revenue of the google store visitors:

1. JSON parsing

As mentioned before there are some JSON fields in our dataset, so we need to flatten them for better handling and visualization but specially because our target is inside of one of those JSON fields.

2. EDA [3]

We will use pandas and matplotlib to get familiar with the data. Checking what type of data we are handling, how many fields are actually important for our goal, in order to give us an idea of how the data should be preprocessed later on , which data we will need to get rid off and how much data is missing and needs to be dealt with.

3. Feature engineering[4]

We will get rid off of features that are useless to our prediction goal, we will create some new ones from what we already have, specially creating new columns from the date feature and the means of certain other features per day, like transaction revenue and number of visits.

4. Data preprocessing.

We will encode the categorical ones using label encoding, handle the missing data either using the mean or filling it with 0s and of course getting the natural log of our Target.

Finally we will create a variable for our Target y and our features X.

5. Data division into training and testing

We will use scikit learn to divide our dataset into 80 % training and the rest into testing.

6. Model testing and choosing

We willing try several ML regression algorithms and score them using the RMSE as our metric.

7. Model benchmarking.

We will train our algorithm using a ‘naive’ version of it and use the scoring results (RMSE) as a benchmark to improving our model.

8. Model tuning using bayesian optimization.

We will use Bayesian Optimization[5] to find the best hyperparameters for our model and see how much we can improve it. Given the technical limitation of my computer this seems like a better option to Grid Search.

9. Model predicting and submitting.

We will use our finan tuned model to predict the revenue in the testing set and submit the file.

REFERENCES.

[1] https://en.wikipedia.org/wiki/Pareto_principle , Pareto principle (20/80 rule) .

[2] https://www.kaggle.com/c/ga-customer-revenue-prediction/data , Google Analytics Customer Revenue Prediction dataset.

[3] https://en.wikipedia.org/wiki/Exploratory_data_analysis, EDA.

[4] https://en.wikipedia.org/wiki/Feature_engineering, Feature Engineering.

[5]https://github.com/fmfn/BayesianOptimization, Bayesian Optimization.